Monitoring the Grid with the Globus Toolkit MDS4

the globus alliance www.globus.org

The Globus Toolkit Monitoring and Discovery System (MDS4) defines and implements mechanisms for service and resource discovery and monitoring in distributed environments. MDS4 is distinguished from previous similar systems by its extensive use of interfaces and behaviors defined in the WS-Resource Framework and WS-Notification specifications, and by its deep integration into essentially every component of the Globus Toolkit.

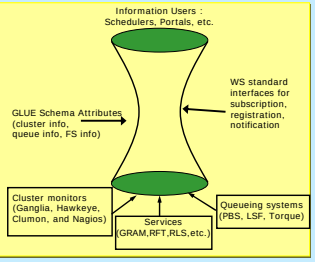
Web Service Standards

MDS4 can be viewed as an exemplary use case for the WS-Resource Framework and WS-Notification specifications. These standards define interfaces for specifying and interacting with data about services. In particular:

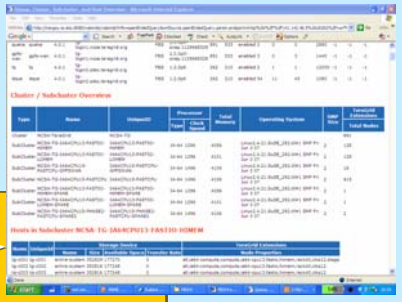
- WS-ResourceProperties defines a mechanism by which Web services can describe and publish resource properties, or sets of information about a resource. Resource property types are defined in the service's WSDL, and resource properties can be retrieved, in the form of XML documents, using WS-ResourceProperties query operations.
- WS-BaseNotification defines a subscription/notification interface for accessing resource property information.
- WS-ServiceGroup defines a mechanism for grouping related resources and/or services together as service groups.

Protocol Hourglass

The neck of the MDS4 "protocol hourglass" (shown right) comprises not only the Web service standard protocols for data access and delivery but also standard schemas for information representation, such as the GLUE schema. Below the neck of the hourglass, MDS4 interfaces to different local information sources, translating their diverse schemas into appropriate XML schema transmitted over WS-RF/WS-N protocols. Above the neck of the hourglass, various tools and applications can take advantage of the uniform Web services query, subscription, and notification interfaces to those information sources that MDS4 implements.



WebMDS uses standard resource property requests to query resource property data and XSLT transforms to format and display them. In this way, we obtain user-friendly front-end to Index data. Web site administrators can customize their own WebMDS deployments by using HTML form options and creating their own XSLT transforms.



The *Trigger service* collects information and compares that data against a set of conditions defined in a configuration file. When a condition is met, an action takes place, such as emailing a system administrator when the disk space on a server reaches a threshold.

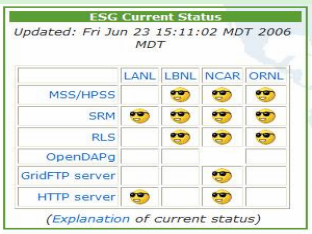
Information Providers, shown here as purple boxes, supply data to the higher level services. These can be anything from simple scripts to resource properties publicized by Web service applications to file scraping.

The *Index service*, shown as yellow ovals, collects information about Grid resources and makes this data available as resource properties to users, clients or other Indexes. A Grid will typically operate multiple Indexes that maintain different data for different purposes.

An *Index server* can also be set up to contain only project specific data, such as the one to the left for Project X, to which the three RFT services have been registered.

Current Deployment
Troubleshooting the Earth Systems Grid

ESG uses MDS4 to monitor the functionality of 17 services at 7 different sites across the US. Information providers are polled every 10 minutes, and when failures occur the Trigger service automatically sends mail to the system administrators so problems can be resolved before they affect users. Because a model of the project's global state is achieved this way, higher-level failures can also be diagnosed. ESG also has this information publicized through a Portal, shown right.



Current Deployment
Resource Discovery for the TeraGrid

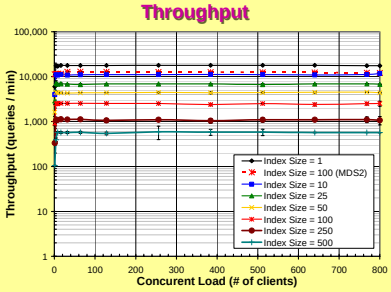
TeraGrid uses MDS4 as an interface to the heterogeneous cluster and queuing systems deployed on the 11 Resource Provider sites. Each local system has made their own deployment decision, but a single interface to the data is needed so users and MetaSchedulers can select the best resource to run a job.

Currently being deployed, this system will eventually collect approximately 30 pieces of data from each of the sites, including basic queuing information, service deployment data, cluster and subcluster data, and static configuration information. This data will then be access either through Java, C, or command line interfaces, or viewed online using a WebMDS interface, such as the one shown above.

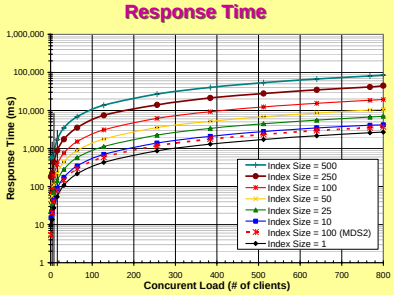
Performance Results

We ran our experiments on the University of Chicago IA32 TeraGrid machine, with 20 client nodes and one server node, connected via 1Gb/s Ethernet. The Index was populated with small sample entries. Each query retrieved all data in the Index. We compare our performance to that of the MDS2 Index service. MDS2 is an LDAP-based monitoring system that is part of the pre-WS Globus Toolkit v 2.4.3. We used a 100-entry Index, which was approximately 11.4KB in size. The queries were for the full Index contents.

Throughput



Response Time



Stability

Vers.	Index Size	Time up (Days)	Queries Processed	Query Per Sec.	Round-trip Time (ms)
4.0.1	25	66+	81,701,925	14	69
4.0.1	50	66+	49,306,104	8	115
4.0.1	100	33	14,686,638	5	194
4.0.0	1	14	93,890,248	76	13
4.0.0	1	96	623,395,877	74	13

The MDS Team is:
Jennifer M. Schopf, ANL/NeSC
Laura Pearlman, ISI
Neill Miller, UC
Carl Kesselman, ISI
Ian Foster, ANL/UC
Mike D'Arcy, ISI

Additional help from:
Eric Blau, ANL/UC
John Bresnahan, UC/ANL
Ann Chevernak ISI
Mike Link, ANL/UC
Ioan Raicu, UC
Mei-Hu Su, ISI
Xuehai Zhang, UC

For further information, contact:
Jennifer M. Schopf
Argonne National Lab and UK NeSC
jms@mcs.anl.gov

This work was supported in part by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under contract W-31-109-Eng-38. Additional support was provided by NSF NMI Award SCI-0438372.